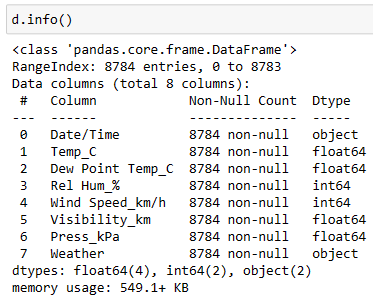
**Report on weather**



The Data Frame contains 8784 entries (rows) with a Range Index from 0 to 8783.

There are a total of 8 columns in the Data Frame, each with a specific name and data type:

Date/Time: This column contains date and time information. It has 8784 non-null entries and is of data type 'object'. To work with date and time data effectively, you might consider converting it to a date time data type using p d .to\_ date time.

Temp\_C: This column represents temperature in degrees Celsius. It has 8784 non-null entries and is of data type 'float64', indicating that it stores decimal numbers.

Dew Point Temp\_C: This column represents the dew point temperature in degrees Celsius. It has 8784 non-null entries and is also of data type 'float64'.

Rel Hum\_%: This column represents relative humidity as a percentage. It has 8784 non-null entries and is of data type 'int64', indicating that it stores integer values.

Wind Speed\_km/h: This column represents wind speed in kilometers per hour. It has 8784 non-null entries and is of data type 'int64'.

Visibility\_km: This column represents visibility in kilometers. It has 8784 non-null entries and is of data type 'float64'.

Press\_kPa: This column represents atmospheric pressure in kilopascals. It has 8784 non-null entries and is of data type 'float64'.

Weather: This column contains information about the weather conditions. It has 8784 non-null entries and is of data type 'object', which typically indicates string or categorical data.

Overall, this DataFrame appears to contain weather-related data, including temperature, humidity, wind speed, visibility, atmospheric pressure, and weather condition descriptions, with each column appropriately labeled and data types assigned. Depending on your analysis or tasks, you can perform various operations and calculations on this data using Pandas.

Top of Form

The DataFrame contains 8784 entries (rows) with a RangeIndex from 0 to 8783.

There are a total of 8 columns in the DataFrame, each with a specific name and data type:

Date/Time: This column contains date and time information. It has 8784 non-null entries and is of data type 'object'. To work with date and time data effectively, you might consider converting it to a datetime data type using pd.to\_datetime.

Temp\_C: This column represents temperature in degrees Celsius. It has 8784 non-null entries and is of data type 'float64', indicating that it stores decimal numbers.

Dew Point Temp\_C: This column represents the dew point temperature in degrees Celsius. It has 8784 non-null entries and is also of data type 'float64'.

Rel Hum\_%: This column represents relative humidity as a percentage. It has 8784 non-null entries and is of data type 'int64', indicating that it stores integer values.

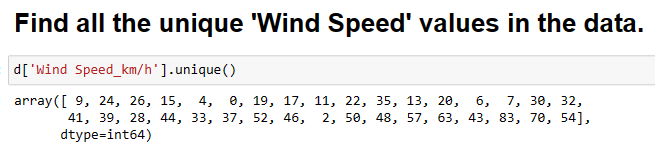
Wind Speed\_km/h: This column represents wind speed in kilometers per hour. It has 8784 non-null entries and is of data type 'int64'.

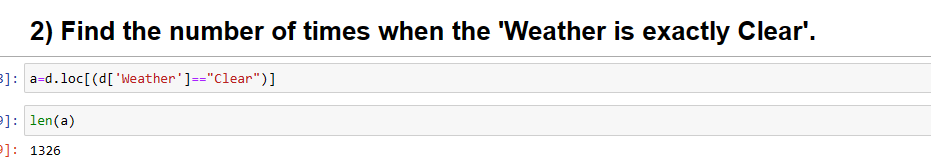
Visibility\_km: This column represents visibility in kilometers. It has 8784 non-null entries and is of data type 'float64'.

Press\_kPa: This column represents atmospheric pressure in kilopascals. It has 8784 non-null entries and is of data type 'float64'.

Weather: This column contains information about the weather conditions. It has 8784 non-null entries and is of data type 'object', which typically indicates string or categorical data.

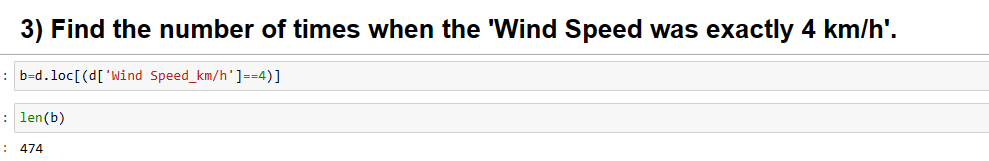
Overall, this DataFrame appears to contain weather-related data, including temperature, humidity, wind speed, visibility, atmospheric pressure, and weather condition descriptions, with each column appropriately labeled and data types assigned. Depending on your analysis or tasks, you can perform various operations and calculations on this data using Pandas.

It shows the unquie ‘Wind Speed’ values from the column “wind speed” from data



d.loc[]: This is used to select rows from the DataFrame based on a condition. In this case, it's selecting rows where the condition d['Weather'] == "Clear" is True.

They are 1326

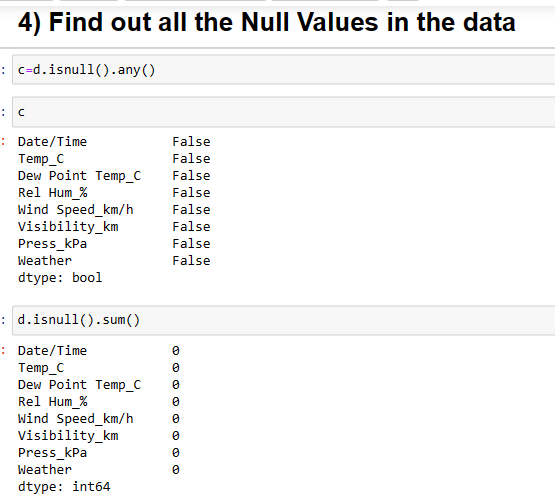


From the data they are 474 times the WindSpeed is exactly 4

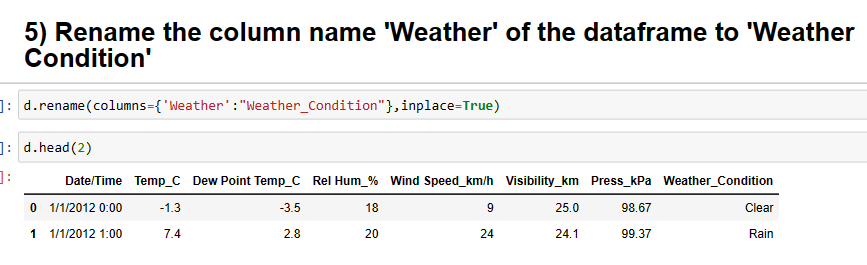
d['Wind Speed\_km/h'] selects the 'Wind Speed\_km/h' column from the DataFrame d.

d['Wind Speed\_km/h'] == 4 creates a Boolean Series where each element is True if the corresponding element in the 'Wind Speed\_km/h' column is equal to 4, and False otherwise.

d.loc[] is used for label-based indexing in pandas DataFrames. It allows you to select rows and columns by labels.

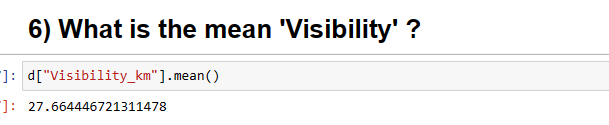


The code d.isnull().sum() is used to count the number of missing (null) values in each column of a DataFrame d.



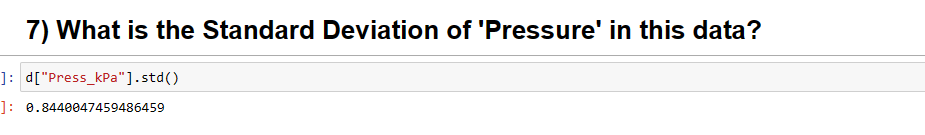
The code you provided is used to rename a column in a DataFrame d

After executing this code with inplace=True, the 'Weather' column in the DataFrame d will be renamed to 'Weather\_Condition', and you can use the updated column name to access or manipulate the data in that column.



It computes the sum of all the values in the 'Visibility\_km' column and then divides that sum by the total number of non-null values in the column. This gives you the average visibility in kilometers across all the rows in the DataFrame d.

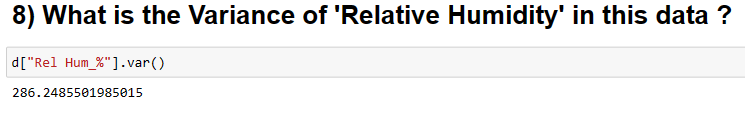
The result of this calculation is a single floating-point number representing the mean visibility in kilometers for the entire dataset.



The Standard Deviation of “pressure” in this data is 0.844

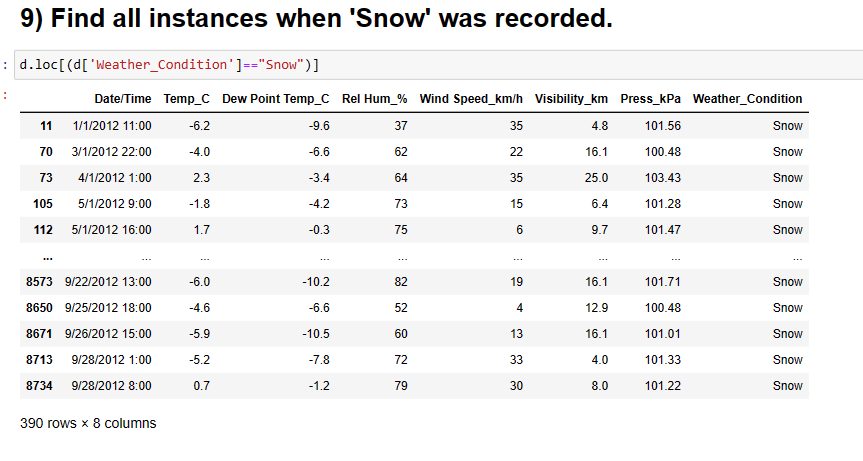
d[“Press\_kpa”] the code extracts the column labeled "Press\_kPa" from the DataFrame d. It assumes that d is a DataFrame or a Series, and "Press\_kPa" is one of its columns.

.std(): This is a Pandas DataFrame/Series method that calculates the standard deviation of the values in the selected column. It computes a measure of the amount of variation or dispersion in the data. The result is a single numerical value representing the standard deviation of the data in the "Press\_kPa" column.



d["Rel Hum\_%"]: This part of the code extracts the column labeled "Rel Hum\_%" from the DataFrame d. It assumes that d is a DataFrame or a Series, and "Rel Hum\_%" is one of its columns.

.var(): This is a Pandas DataFrame/Series method that calculates the variance of the values in the selected column. Variance measures how much the values in the data set vary from the mean (average). It's a measure of the data's spread or dispersion. The result is a single numerical value representing the variance of the data in the "Rel Hum\_%" column.

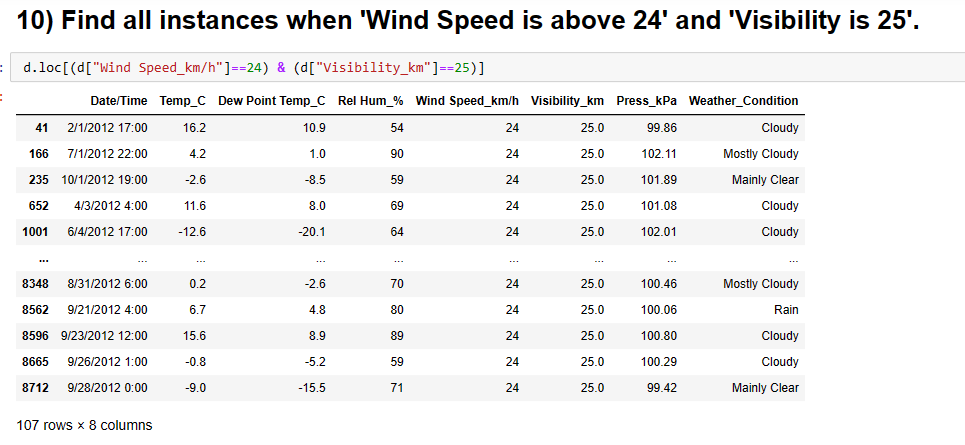


They are 390 data points where the weather condition is snow

d['Weather\_Condition']: This part of the code extracts the values from the "Weather\_Condition" column of the DataFrame d. It assumes that d is a DataFrame with a column named "Weather\_Condition."

(d['Weather\_Condition']=="Snow"): This part of the code creates a Boolean mask that checks if each element in the "Weather\_Condition" column is equal to "Snow." It returns a Boolean Series where each element is True if the condition is met and False otherwise.

d.loc[...]: Finally, the .loc method is used to locate the rows in the DataFrame d where the condition is True. As a result, it selects all the rows where the weather condition is "Snow" and returns them as a new DataFrame.



They are 107 data points where ‘Wind Speed\_Km/h’ is equal to 24 and ‘Visibility\_km’ is equal to 25

d['Wind Speed\_km/h']: This part of the code extracts the values from the "Wind Speed\_km/h" column of the DataFrame d. It assumes that d is a DataFrame with a column named "Wind Speed\_km/h."

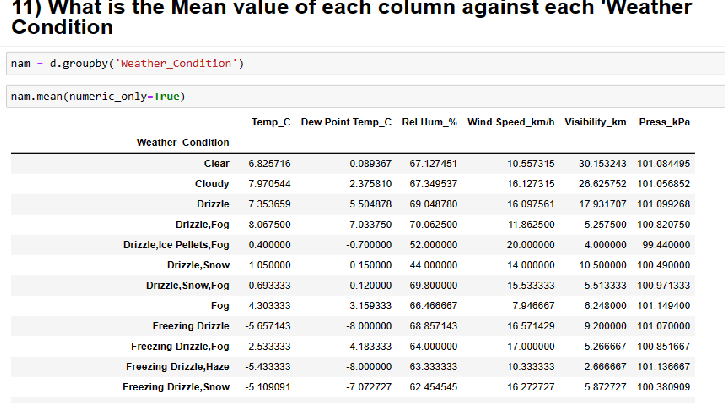
d['Visibility\_km']: This part of the code extracts the values from the "Visibility\_km" column of the DataFrame d. It assumes that d is a DataFrame with a column named "Visibility\_km."

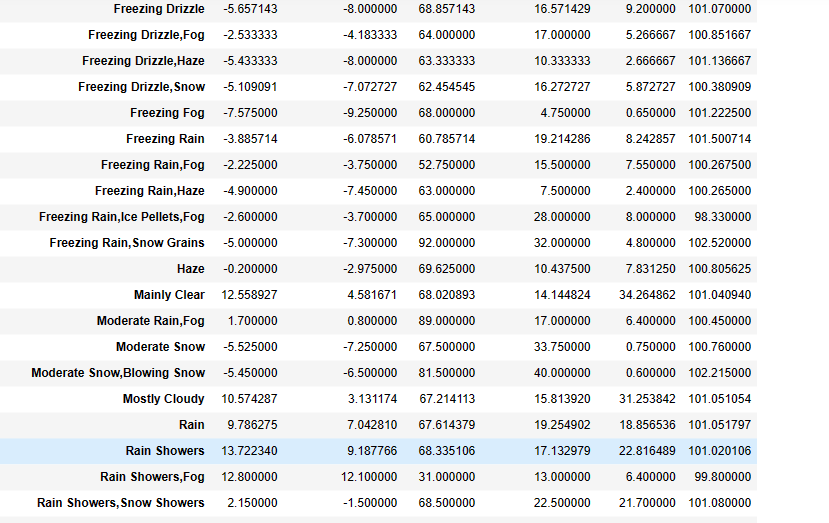
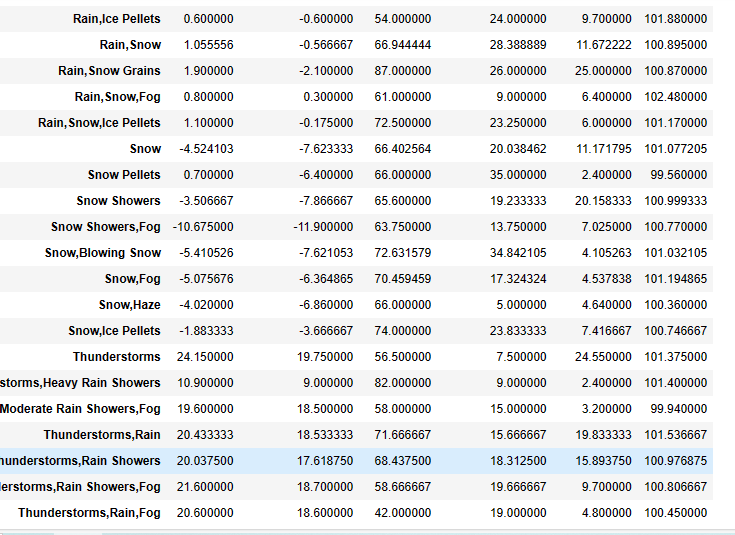
(d["Wind Speed\_km/h"]==24): This part of the code creates a Boolean mask that checks if each element in the "Wind Speed\_km/h" column is equal to 24. It returns a Boolean Series where each element is True if the condition is met and False otherwise.

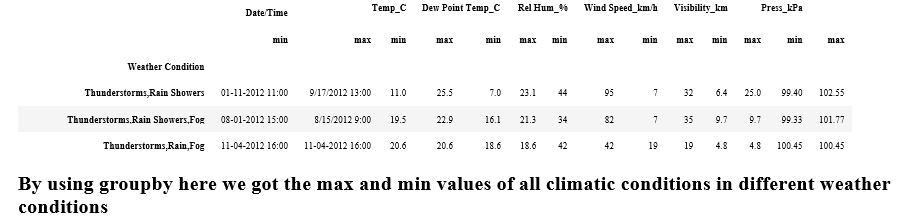
(d["Visibility\_km"]==25): This part of the code creates a Boolean mask that checks if each element in the "Visibility\_km" column is equal to 25. It returns a Boolean Series where each element is True if the condition is met and False otherwise.

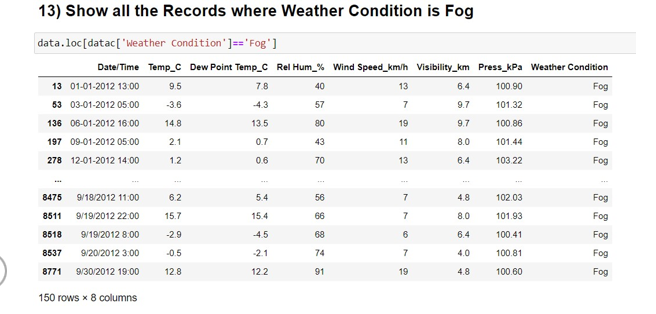
&: The & operator is used to combine these two Boolean Series using the logical AND operation, which means it returns True only when both conditions are True.

d.loc[...]: Finally, the .loc method is used to locate the rows in the DataFrame d where the combined condition is True. It selects all the rows where both the wind speed is 24 km/h and the visibility is 25 km, and it returns them as a new DataFrame

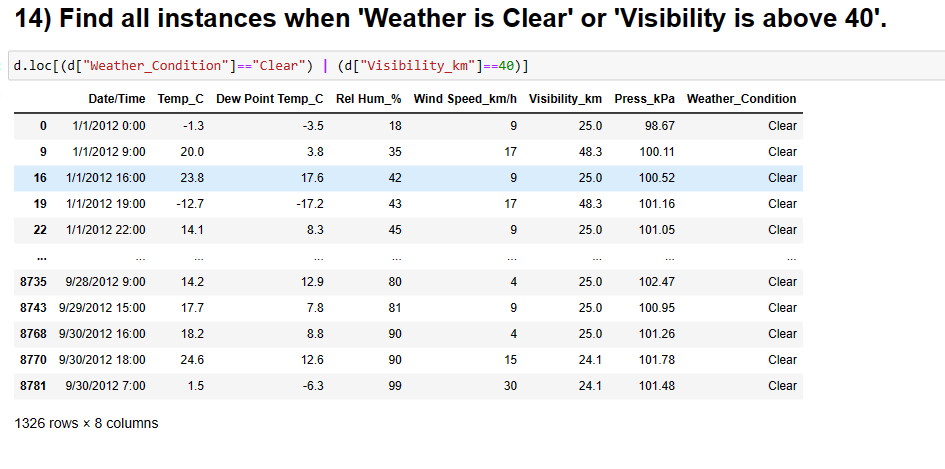




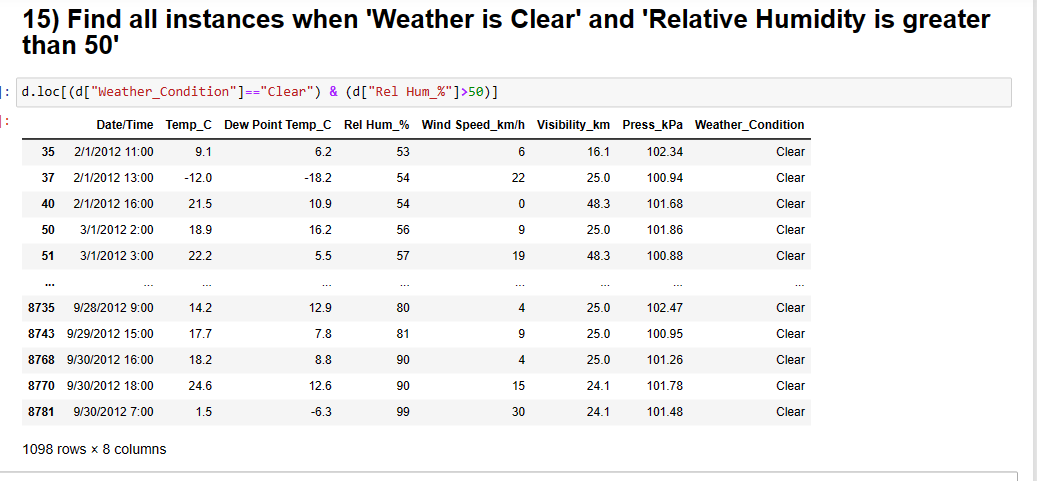




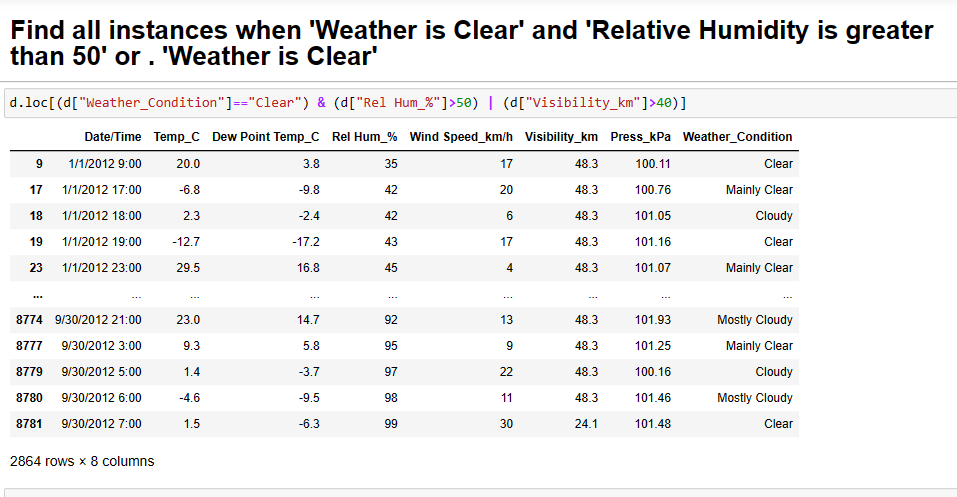
The Total data points where “weather condition” is equal to Fog are 150



These are all instance when “weather is clear “ or “visibility is above 40”



They are 1098 instances when “weather is clear” and “relative humidity is greater than 50”



These are instances when ‘weather is clear ‘ and ‘relative humidity is greater than 50’ or ‘weather is clear’